
SQL 활용

정기 수행 평가 05



Contents

1

프로젝트 생성

- 프로젝트 구조

2

화면 구현

- book 화면 구현
- customer 화면 구현

3

기능 구현

- DBCP 커넥션
- book, customer CRUD 구현

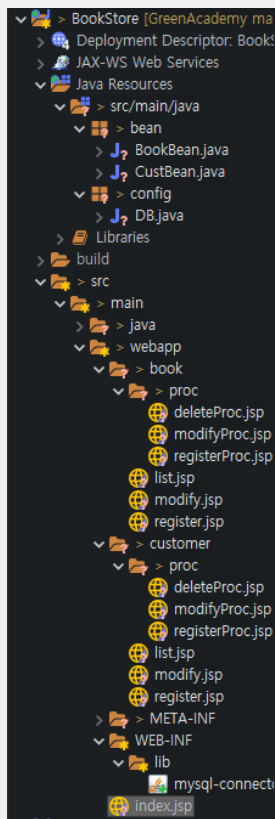
4

실행

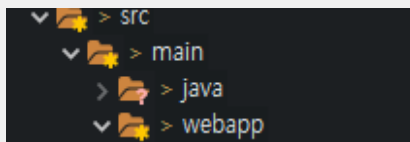
- 실제 동작 화면

1 프로젝트 생성

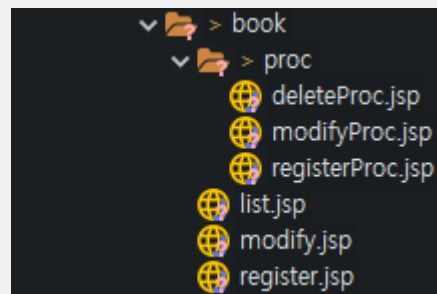
| 프로젝트 구조



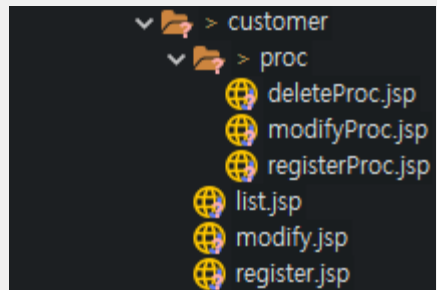
DB 관련 클래스들



DB 라이브러리 및 메인화면



도서 화면 및 기능



고객 화면 및 기능

2 화면 구현

| book 화면 구현

BookStore

도서목록 고객목록

도서목록

처음으로 도서등록

| 도서번호 | 도서명 | 출판사 | 가격 | 관리 |
|------|-------------------|---------|-------|-------|
| 1 | 죽구의 역사 | 굿스포츠 | 7000 | 수정 삭제 |
| 2 | 죽구아는 여자 | 나무수 | 13000 | 수정 삭제 |
| 3 | 죽구의 이해 | 대한미디어 | 22000 | 수정 삭제 |
| 4 | 골프 바이블 | 대한미디어 | 35000 | 수정 삭제 |
| 5 | 피겨 교본 | 굿스포츠 | 8000 | 수정 삭제 |
| 6 | 역도 단계별기술 | 굿스포츠 | 6000 | 수정 삭제 |
| 7 | 야구의 추억 | 이상미디어 | 20000 | 수정 삭제 |
| 8 | 야구를 부탁해 | 이상미디어 | 13000 | 수정 삭제 |
| 9 | 올림픽 이야기 | 삼성당 | 7500 | 수정 삭제 |
| 10 | Olympic Champions | Pearson | 13000 | 수정 삭제 |

도서등록

처음으로 도서등록

| | |
|-----------------------------------|----------------------|
| 도서번호 | <input type="text"/> |
| 도서명 | <input type="text"/> |
| 출판사 | <input type="text"/> |
| 가격 | <input type="text"/> |
| <input type="button" value="등록"/> | |

도서수정

처음으로 도서등록

| | |
|-----------------------------------|--------|
| 도서번호 | 1 |
| 도서명 | 죽구의 역사 |
| 출판사 | 굿스포츠 |
| 가격 | 7000 |
| <input type="button" value="수정"/> | |

```
<body>
<h3>BookStore</h3>

<a href="/BookStore/book/list.jsp">도서목록</a>
<a href="/BookStore/customer/list.jsp">고객목록</a>

</body>
```

```
<body>
<h3>도서목록</h3>

<a href="/BookStore/index.jsp">처음으로</a>
<a href="/BookStore/book/register.jsp">도서등록</a>

<table border="1">
<tr>
<th>도서번호</th>
<th>도서명</th>
<th>출판사</th>
<th>가격</th>
<th>관리</th>
</tr>
<% for(BookBean bb : books){ %>
<tr>
<td><%= bb.getBookId() %></td>
<td><%= bb.getBookName() %></td>
<td><%= bb.getPublisher() %></td>
<td><%= bb.getPrice() %></td>
<td>
<a href="/BookStore/book/modify.jsp?bookId=<%= bb.getBookId() %>">수정</a>
<a href="/BookStore/book/proc/deleteProc.jsp?bookId=<%= bb.getBookId() %>">삭제</a>
</td>
</tr>
<% } %>
</table>
</body>
```

```
<body>
<h3>도서등록</h3>

<a href="/BookStore/index.jsp">처음으로</a>
<a href="/BookStore/book/list.jsp">도서목록</a>

<form action="/BookStore/book/proc/registerProc.jsp"
<table border="1">
<tr>
<td>도서번호</td>
<td><input type="text" name="bookId" /> </td>
</tr>
<tr>
<td>도서명</td>
<td><input type="text" name="bookName" /> </td>
</tr>
<tr>
<td>출판사</td>
<td><input type="text" name="publisher" /> </td>
</tr>
<tr>
<td>가격</td>
<td><input type="text" name="price" /> </td>
</tr>
<tr>
<td colspan="2" align="right"><input type="submit" value="등록" /> </td>
</tr>
</table>
</form>
</body>
```

```
<body>
<h3>도서수정</h3>

<a href="/BookStore/index.jsp">처음으로</a>
<a href="/BookStore/book/list.jsp">도서목록</a>

<form action="/BookStore/book/proc/modifyProc.jsp"
<table border="1">
<tr>
<td>도서번호</td>
<td><input type="text" name="bookId" readonly value="<%=bb.getBookId()%>" /> </td>
</tr>
<tr>
<td>도서명</td>
<td><input type="text" name="bookName" value="<%=bb.getBookName()%>" /> </td>
</tr>
<tr>
<td>출판사</td>
<td><input type="text" name="publisher" value="<%=bb.getPublisher()%>" /> </td>
</tr>
<tr>
<td>가격</td>
<td><input type="text" name="price" value="<%=bb.getPrice()%>" /> </td>
</tr>
<tr>
<td colspan="2" align="right"><input type="submit" value="수정" /> </td>
</tr>
</table>
</form>
</body>
```

2 화면 구현

| customer 화면 구현

BookStore

[도서목록](#) [고객목록](#)

고객목록

처음으로 [신규등록](#)

| 고객번호 | 고객명 | 주소 | 휴대폰 | 관리 |
|------|-----|----------|---------------|---------------------------------------|
| 1 | 박지성 | 영국 맨체스터 | 000-5000-0001 | 수정 삭제 |
| 2 | 김연아 | 대한민국 서울 | 000-6000-0001 | 수정 삭제 |
| 3 | 장미란 | 대한민국 강원도 | 000-7000-0001 | 수정 삭제 |
| 4 | 추신수 | 미국 클리블랜드 | 000-8000-0001 | 수정 삭제 |
| 5 | 박세리 | 대한민국 대전 | null | 수정 삭제 |

고객등록

처음으로 [도서목록](#)

고객명

주소

휴대폰

고객수정

처음으로 [고객목록](#)

고객번호

고객명

주소

휴대폰

```
<body>
  <h3>BookStore</h3>

  <a href="/BookStore/book/list.jsp">도서목록</a>
  <a href="/BookStore/customer/list.jsp">고객목록</a>

</body>
```

```
<body>
  <h3>고객등록</h3>

  <a href="/BookStore/index.jsp">처음으로</a>
  <a href="/BookStore/customer/list.jsp">도서목록</a>

  <form action="/BookStore/customer/proc/registerProc.jsp">
    <table border="1">
      <tr>
        <td>고객명</td>
        <td><input type="text" name="custName" /></td>
      </tr>
      <tr>
        <td>주소</td>
        <td><input type="text" name="custAddr" /></td>
      </tr>
      <tr>
        <td>휴대폰</td>
        <td><input type="text" name="custHp" /></td>
      </tr>
      <tr>
        <td colspan="2" align="right"><input type="submit" value="등록" /></td>
      </tr>
    </table>
  </form>
</body>
```

```
<body>
  <h3>고객목록</h3>

  <a href="/BookStore/index.jsp">처음으로</a>
  <a href="/BookStore/customer/register.jsp">고객등록</a>

  <table border="1">
    <tr>
      <th>고객번호</th>
      <th>고객명</th>
      <th>주소</th>
      <th>휴대폰</th>
      <th>관리</th>
    </tr>
    <% for(CustBean cb : customers){ %>
    <tr>
      <td><%= cb.getCustId() %></td>
      <td><%= cb.getCustName() %></td>
      <td><%= cb.getCustAddr() %></td>
      <td><%= cb.getCustHp() %></td>
      <td>
        <a href="/BookStore/customer/modify.jsp?custId=<%= cb.getCustId() %>">수정</a>
        <a href="/BookStore/customer/proc/deleteProc.jsp?custId=<%= cb.getCustId() %>">삭제</a>
      </td>
    </tr>
    <% } %>
  </table>
</body>
```

```
<body>
  <h3>고객수정</h3>

  <a href="/BookStore/index.jsp">처음으로</a>
  <a href="/BookStore/customer/list.jsp">고객목록</a>

  <form action="/BookStore/customer/proc/modifyProc.jsp">
    <table border="1">
      <tr>
        <td>고객번호</td>
        <td><input type="text" name="custId" readonly value="<%=cb.getCustId()%>" /></td>
      </tr>
      <tr>
        <td>고객명</td>
        <td><input type="text" name="custName" value="<%=cb.getCustName()%>" /></td>
      </tr>
      <tr>
        <td>주소</td>
        <td><input type="text" name="custAddr" value="<%=cb.getCustAddr()%>" /></td>
      </tr>
      <tr>
        <td>휴대폰</td>
        <td><input type="text" name="custHp" value="<%=cb.getCustHp()%>" /></td>
      </tr>
      <tr>
        <td colspan="2" align="right"><input type="submit" value="수정" /></td>
      </tr>
    </table>
  </form>
</body>
```

3 기능 구현

| DBCP 커넥션

```
//Java1_BookStore
<Resource
    name="regularExam_java1_bookstore"
    auth="Container"
    type="javax.sql.DataSource"
    driverClassName="com.mysql.cj.jdbc.Driver"
    url="jdbc:mysql://127.0.0.1:3306/java1_bookstore"
    username="java1_book"
    password="1234"
    maxTotal="8"
    maxIdle="8"
    maxWaitMillis="3000"
/>
```

톰캣 서버 context.xml에 커넥션 풀 생성

```
public class DB {

    private static DataSource ds = null;

    public static Connection getConnection() throws NamingException, SQLException {
        if(ds == null) {
            ds = (DataSource) new InitialContext().lookup("java:comp/env/regularExam_java1_bookstore");
        }
        return ds.getConnection();
    }

}
```

bookstore 전용 DB 연결 클래스 생성

```
Connection conn = DB.getConnection();
String sql = "DELETE FROM `customer` WHERE `custId`=?";
PreparedStatement psmt = conn.prepareStatement(sql);
psmt.setInt(1, custId);
psmt.executeUpdate();

psmt.close();
conn.close();
```

실사용

3

기능 구현

| book, customer CRUD 구현

```
//cust DB처리
List<CustBean> customers = null;
try{
    Connection conn = DB.getConnection();
    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM `customer`");

    customers = new ArrayList<>();

    while(rs.next()){
        CustBean cb = new CustBean();

        cb.setCustId(rs.getInt(1));
        cb.setCustName(rs.getString(2));
        cb.setCustAddr(rs.getString(3));
        cb.setCustHp(rs.getString(4));

        customers.add(cb);
    }
    rs.close();
    stmt.close();
    conn.close();
}
```

1. DB클래스로 DB와 연결 후
2. Statement로 Select SQL문 실행
3. 가져온 목록값을 ResultSet에 저장
4. Bean에 각 필드값을 저장
5. 배열에 Bean 추가

```
<% for(CustBean cb : customers){ %>
<tr>
    <td><%= cb.getCustId() %></td>
    <td><%= cb.getCustName() %></td>
    <td><%= cb.getCustAddr() %></td>
    <td><%= cb.getCustHp() %></td>
    <td>
        <a href="/BookStore/customer/modify.jsp?custId=<%= cb.getCustId() %>">수정</a>
        <a href="/BookStore/customer/proc/deleteProc.jsp?custId=<%= cb.getCustId() %>">삭제</a>
    </td>
</tr>
<% } %>
```

1. 서블릿을 이용해 For문으로 테이블 생성
2. 수정과 삭제를 위해 Get방식으로 Id값을 전송

3 기능 구현

| book, customer CRUD 구현

```
request.setCharacterEncoding("utf-8");
int custId = Integer.parseInt(request.getParameter("custId"));

BookBean bb = null;
try{
    Connection conn = DB.getConnection();
    String sql = "DELETE FROM `customer` WHERE `custId`=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setInt(1, custId);
    pstmt.executeUpdate();
}
```

deleteProc 주요 코드

```
request.setCharacterEncoding("utf-8");

String custName = request.getParameter("custName");
String custAddr = request.getParameter("custAddr");
String custHp = request.getParameter("custHp");

try{
    Connection conn = DB.getConnection();

    String sql = "INSERT INTO `customer` SET `custName`=?, `custAddr`=?, `custHp`=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);

    pstmt.setString(1, custName);
    pstmt.setString(2, custAddr);
    pstmt.setString(3, custHp);

    pstmt.executeUpdate();
}
```

registerProc 주요 코드

```
request.setCharacterEncoding("utf-8");

int custId = Integer.parseInt(request.getParameter("custId"));
String custName = request.getParameter("custName");
String custAddr = request.getParameter("custAddr");
String custHp = request.getParameter("custHp");

try{
    Connection conn = DB.getConnection();

    String sql = "UPDATE `customer` SET `custName`=?, `custAddr`=?, `custHp`=? ";
    sql += "WHERE `custId`=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);

    pstmt.setString(1, custName);
    pstmt.setString(2, custAddr);
    pstmt.setString(3, custHp);
    pstmt.setInt(4, custId);

    pstmt.executeUpdate();
}
```

modifyProc 주요 코드

```
pstmt.close();
conn.close();
}
catch(Exception e){
    e.printStackTrace();
}

response.sendRedirect("/BookStore/customer/list.jsp");
```

작업 완료 후 연결 해제 및 목록 이동

4 실행

| 실제 동작 화면

고객목록

처음으로 고객등록

| 고객번호 | 고객명 | 주소 | 휴대폰 | 관리 |
|------|-----|----------|---------------|---------------------------------------|
| 1 | 박지성 | 영국 맨체스타 | 000-5000-0001 | 수정 삭제 |
| 2 | 김연아 | 대한민국 서울 | 000-6000-0001 | 수정 삭제 |
| 3 | 장미란 | 대한민국 강원도 | 000-7000-0001 | 수정 삭제 |
| 4 | 추신수 | 미국 클리블랜드 | 000-8000-0001 | 수정 삭제 |
| 5 | 박세리 | 대한민국 대전 | null | 수정 삭제 |

고객등록

처음으로 도서목록

| | |
|-----------------------------------|---------------|
| 고객명 | Test1 |
| 주소 | Test1 |
| 휴대폰 | 010-test-test |
| <input type="button" value="등록"/> | |

고객목록

처음으로 고객등록

| 고객번호 | 고객명 | 주소 | 휴대폰 | 관리 |
|------|-------|----------|---------------|---------------------------------------|
| 1 | 박지성 | 영국 맨체스타 | 000-5000-0001 | 수정 삭제 |
| 2 | 김연아 | 대한민국 서울 | 000-6000-0001 | 수정 삭제 |
| 3 | 장미란 | 대한민국 강원도 | 000-7000-0001 | 수정 삭제 |
| 4 | 추신수 | 미국 클리블랜드 | 000-8000-0001 | 수정 삭제 |
| 5 | 박세리 | 대한민국 대전 | null | 수정 삭제 |
| 7 | Test1 | Test1 | 010-test-test | 수정 삭제 |

고객수정

처음으로 고객등록

| | |
|-----------------------------------|---------------|
| 고객번호 | 7 |
| 고객명 | Test |
| 주소 | Test1 |
| 휴대폰 | 010-1234-1234 |
| <input type="button" value="수정"/> | |

고객목록

처음으로 고객등록

| 고객번호 | 고객명 | 주소 | 휴대폰 | 관리 |
|------|------|----------|---------------|---------------------------------------|
| 1 | 박지성 | 영국 맨체스타 | 000-5000-0001 | 수정 삭제 |
| 2 | 김연아 | 대한민국 서울 | 000-6000-0001 | 수정 삭제 |
| 3 | 장미란 | 대한민국 강원도 | 000-7000-0001 | 수정 삭제 |
| 4 | 추신수 | 미국 클리블랜드 | 000-8000-0001 | 수정 삭제 |
| 5 | 박세리 | 대한민국 대전 | null | 수정 삭제 |
| 7 | Test | Test1 | 010-1234-1234 | 수정 삭제 |

고객목록

처음으로 고객등록

| 고객번호 | 고객명 | 주소 | 휴대폰 | 관리 |
|------|------|----------|---------------|---------------------------------------|
| 1 | 박지성 | 영국 맨체스타 | 000-5000-0001 | 수정 삭제 |
| 2 | 김연아 | 대한민국 서울 | 000-6000-0001 | 수정 삭제 |
| 3 | 장미란 | 대한민국 강원도 | 000-7000-0001 | 수정 삭제 |
| 4 | 추신수 | 미국 클리블랜드 | 000-8000-0001 | 수정 삭제 |
| 7 | Test | Test1 | 010-1234-1234 | 수정 삭제 |

도서목록

처음으로 도서등록

| 도서번호 | 도서명 | 출판사 | 가격 | 관리 |
|------|-------------------|---------|-------|---------------------------------------|
| 1 | 축구의 역사 | 굿스포츠 | 7000 | 수정 삭제 |
| 2 | 축구하는 여자 | 나무수 | 13000 | 수정 삭제 |
| 3 | 축구의 이해 | 대한미디어 | 22000 | 수정 삭제 |
| 4 | 골프 바이블 | 대한미디어 | 35000 | 수정 삭제 |
| 5 | 피겨 교본 | 굿스포츠 | 8000 | 수정 삭제 |
| 6 | 역도 단계별기술 | 굿스포츠 | 6000 | 수정 삭제 |
| 7 | 야구의 추억 | 이상미디어 | 20000 | 수정 삭제 |
| 8 | 야구를 부탁해 | 이상미디어 | 13000 | 수정 삭제 |
| 9 | 올림픽 이야기 | 삼성당 | 7500 | 수정 삭제 |
| 10 | Olympic Champions | Pearson | 13000 | 수정 삭제 |

도서등록

처음으로 도서등록

| 도서번호 | 도서명 | 출판사 | 가격 | 관리 |
|------|-------------------|---------|-------|---------------------------------------|
| 1 | 축구의 역사 | 굿스포츠 | 7000 | 수정 삭제 |
| 2 | 축구하는 여자 | 나무수 | 13000 | 수정 삭제 |
| 9 | 올림픽 이야기 | 삼성당 | 7500 | 수정 삭제 |
| 10 | Olympic Champions | Pearson | 13000 | 수정 삭제 |

도서등록

처음으로 도서등록

| | |
|-----------------------------------|-------|
| 도서번호 | 3 |
| 도서명 | Hello |
| 출판사 | World |
| 가격 | 8900 |
| <input type="button" value="등록"/> | |

도서목록

처음으로 도서등록

| 도서번호 | 도서명 | 출판사 | 가격 | 관리 |
|------|-------------------|---------|-------|---------------------------------------|
| 1 | 축구의 역사 | 굿스포츠 | 7000 | 수정 삭제 |
| 2 | 축구하는 여자 | 나무수 | 13000 | 수정 삭제 |
| 3 | Hello | World | 8900 | 수정 삭제 |
| 9 | 올림픽 이야기 | 삼성당 | 7500 | 수정 삭제 |
| 10 | Olympic Champions | Pearson | 13000 | 수정 삭제 |

도서수정

처음으로 도서등록

| | |
|-----------------------------------|--------------|
| 도서번호 | 3 |
| 도서명 | Hello World! |
| 출판사 | GA |
| 가격 | 89000 |
| <input type="button" value="수정"/> | |

도서목록

처음으로 도서등록

| 도서번호 | 도서명 | 출판사 | 가격 | 관리 |
|------|-------------------|---------|-------|---------------------------------------|
| 1 | 축구의 역사 | 굿스포츠 | 7000 | 수정 삭제 |
| 2 | 축구하는 여자 | 나무수 | 13000 | 수정 삭제 |
| 3 | Hello World! | GA | 89000 | 수정 삭제 |
| 9 | 올림픽 이야기 | 삼성당 | 7500 | 수정 삭제 |
| 10 | Olympic Champions | Pearson | 13000 | 수정 삭제 |

고객 목록 동작 화면

도서 목록 동작 화면

정기 수행 평가

05.SQL 활용 Book Store



Thank you!

Kim Dong Geun
From GreenAcademy
2022.10.21